

1 **CLAIMS**

2

3 1. A programming interface embodied on one or more computer
4 readable media, comprising:

5 a first group of services related to generating graphical objects;
6 a second group of services related to formatting content; and
7 a third group of services related to creating components of the graphical
8 objects.

9

10 2. A programming interface as recited in claim 1, wherein the first
11 group of services, the second group of services and the third group of services
12 share a common programming model.

13

14 3. A programming interface as recited in claim 1, wherein the first
15 group of services, the second group of services and the third group of services
16 utilize a common markup language.

17

18 4. A programming interface as recited in claim 1, wherein the first
19 group of services, the second group of services and the third group of services
20 share a common event system.

21

22 5. A programming interface as recited in claim 1, wherein the first
23 group of services, the second group of services and the third group of services
24 share a common property definition system.

1 6. A programming interface as recited in claim 1, wherein the first
2 group of services, the second group of services and the third group of services
3 share a common input paradigm.

4

5 7. A programming interface as recited in claim 1, wherein the first
6 group of services, the second group of services and the third group of services
7 share a common system for nesting elements associated with a particular group of
8 services within elements associated with another group of services.

9

10 8. A programming interface as recited in claim 1, wherein the first
11 group of services includes a service that determines an appearance of the graphical
12 objects.

13

14 9. A programming interface as recited in claim 1, wherein the first
15 group of services includes a service that determines a behavior of the graphical
16 objects.

17

18 10. A programming interface as recited in claim 1, wherein the first
19 group of services includes a service that determines an arrangement of the
20 graphical objects.

21

22 11. A programming interface as recited in claim 1, wherein the first
23 group of services includes a plurality of nested elements that define the graphical
24 objects.

1 **12.** A programming interface as recited in claim 1, wherein the
2 graphical objects are comprised of one or more elements defined by vector
3 graphics.
4

5 **13.** A programming interface as recited in claim 1, wherein the first
6 group of services can define window properties in a markup language without
7 launching a new window.
8

9 **14.** A programming interface as recited in claim 1, wherein the first
10 group of services generate a user interface containing a plurality of graphical
11 objects.
12

13 **15.** A programming interface as recited in claim 1, wherein the second
14 group of services arrange the graphical objects.
15

16 **16.** A software architecture comprising the programming interface as
17 recited in claim 1.
18
19
20
21
22
23
24
25

1 17. An application program interface embodied on one or more
2 computer readable media, comprising:

3 a first group of services related to generating graphical objects;
4 a second group of services related to formatting content; and
5 a third group of services related to creating components of the graphical
6 objects, wherein the first group of services, the second group of services and the
7 third group of services share a common programming model.

8

9 18. An application program interface as recited in claim 17, wherein the
10 first group of services, the second group of services and the third group of services
11 utilize a common markup language.

12

13 19. An application program interface as recited in claim 17, wherein the
14 third group of services includes services to generate geometric shapes.

15

16 20. An application program interface as recited in claim 17, wherein the
17 second group of services includes arranging a plurality of data elements.

18

19 21. An application program interface as recited in claim 17, wherein the
20 first group of services includes:

21 a service that determines an appearance of a graphical object; and
22 a service that determines a behavior of the graphical object.

1 **22.** An application program interface as recited in claim 17, wherein the
2 first group of services includes a service that defines window properties in a
3 markup language without launching a new window.

4

5 **23.** A computer system including one or more microprocessors and one
6 or more software programs, the one or more software programs utilizing a
7 programming interface to request services from an operating system, the
8 programming interface including separate commands to request services
9 consisting of the following groups of services:

10 a first group of services for generating graphical objects; and
11 a second group of services for creating components of the graphical objects,
12 wherein the first group of services and the second group of services share a
13 common programming model.

14

15 **24.** A computer system as recited in claim 23, wherein the first group of
16 services includes:

17 a service for defining an appearance of the graphical objects; and
18 a service for defining an arrangement of the graphical objects.

19

20 **25.** A computer system as recited in claim 23, wherein the second group
21 of services includes services to generate a plurality of geometric shapes.

1 **26.** A method comprising:

2 calling one or more first functions to facilitate generating graphical objects;

3 and

4 calling one or more second functions to facilitate formatting content,
5 wherein the first functions and the second functions share a common programming
6 model.

7

8 **27.** A method as recited in claim 26, further including calling one or
9 more third functions to facilitate creating components of the graphical objects.

10

11 **28.** A method as recited in claim 26, further including calling one or
12 more third functions to facilitate generating geometric shapes contained in the
13 graphical objects.

14

15 **29.** A method as recited in claim 26, wherein the first functions
16 facilitate:

17 defining window properties in a markup language without launching a new
18 window; and

19 generating a user interface containing a plurality of graphical objects.

1 **30.** A system comprising:

2 means for exposing a first set of functions that enable generating graphical

3 objects; and

4 means for exposing a second set of functions that enable creating

5 components of the graphical objects, wherein the components of the graphical

6 objects include a plurality of geometric shapes, and wherein the first set of

7 functions and the second set of functions share a common programming model. .

8

9 **31.** A system as recited in claim 30, wherein the second set of functions

10 further enable arrangement of the geometric shapes on a page to be rendered.

11

12 **32.** A system as recited in claim 30, further comprising means for

13 exposing a third set of functions that enable formatting content for display.

14

15 **33.** A system as recited in claim 30, wherein the first set of functions

16 and the second set of functions utilize a common markup language.

17

18 **34.** A system as recited in claim 30, wherein the first set of functions

19 and the second set of functions share a common event system and a common

20 property definition system.